**CSE 1201: Object Oriented Programming  
Md. Mahmudul Hasan**

**What is Object Oriented Programming?**

Object-Oriented Programming (OOP) is a programming paradigm (দৃষ্টান্ত) that depends on the concept of classes and objects.   
It is used to structure a software program into simple, reusable pieces of code blueprints (usually called classes), which are used to create individual instances of objects.   
There are many object-oriented programming languages including JavaScript, C++, Java, and Python.

A class is an abstract blueprint used to create more specific, concrete objects. Classes often represent broad categories, like Car or Dog that share attributes.

Classes can also contain functions, called methods available only to objects of that type. These functions are defined within the class and perform some action helpful to that specific type of object.

**Class:**   
A class in C++ is the building block that leads to Object-Oriented programming. It is a user-defined data type, which holds its own data members and member functions, which can be accessed and used by creating an instance of that class. A C++ class is like a blueprint for an object.

* A Class is a user defined data-type which has data members and member functions.
* Data members are the data variables and member functions are the functions used to manipulate these variables and together these data members and member functions defines the properties and behavior of the objects in a Class.
* In the above example of class Car, the data member will be speed limit, mileage etc and member functions can be applied brakes, increase speed etc.

A class is defined in C++ using keyword class followed by the name of class. The body of class is defined inside the curly brackets and terminated by a semicolon at the end.

Syntax: Class className{ ………………..};

**Object:**

An **Object** is an instance of a Class. When a class is defined, no memory is allocated but when it is instantiated (i.e., an object is created) memory is allocated.

**Declaring Objects:** When a class is defined, only the specification for the object is defined; no memory or storage is allocated. To use the data and access functions defined in the class, you need to create objects.

Syntax: ClassName ObjectName;

**Constructor:**

A constructor is a special type of function that is used to initialize the object. A constructor is a special type of member function that is called automatically when an object is created.

**Constructor Properties:**

* Constructor is a special type of function.
* Constructor has the same name as that of the class it belongs.
* It has no return type not even void.
* It is called automatically.

**Constructors are of three types:**

* Default Constructor
* Parametrized Constructor
* Copy Constructor

**Default constructor:**

A default constructor is a constructor which doesn't take any argument. It has no parameter.

class\_name()

{

// Default constructor

}

**Parameterized Constructors**

These are the constructors with parameter. Using this Constructor you can provide different values to data members of different objects, by passing the appropriate values as argument.

class\_name(int x, double y)

{

// Parameterized Constructors

}

**Copy Constructor**

The copy constructor in C++ is used to copy data of one object to another. A copy constructor is like a normal parameterized Constructor, but in Copy Constructor, the parameter is the class object. In other words, we can say that the Copy constructor uses to initialize an object using another object of a similar class.

There are four basic principles in Object-Oriented programming: Encapsulation, Abstraction, Inheritance and Polymorphism.

**Advantages of OOP**

* Re-usability
* Data Redundancy
* Code Maintenance
* Security
* Design Benefits
* Better productivity
* Easy troubleshooting
* Polymorphism Flexibility
* Problems solving